US-PAT-NO: 6691067

DOCUMENT-IDENTIFIER: US 6691067 B1

TITLE: Enterprise management system and method which includes statistical recreation of system **resource** usage for more accurate **monitoring,** prediction, and performance workload characterization

---------- KWIC ---------

Abstract Text - ABTX (1):

A system and method for estimating statistics concerning system metrics to provide for the accurate and efficient monitoring of one or more computer systems. The system preferably comprises a distributed computing environment, i.e., an enterprise, which comprises a plurality of interconnected computer systems. At least one of the computer systems is an agent computer system which includes agent software and/or system software for the collection of data relating to one or more metrics, i.e., measurements of system resources. Metric data is continually collected over the course of a measurement interval, regularly placed into a registry of metrics, and then periodically sampled from the registry indirectly. Sampling-related uncertainty and inaccuracy arise from two primary sources: the unsampled residual segments of seen (i.e., sampled and therefore known) events, and unseen (i.e., unsampled and therefore unknown) events. The total unsampled **utilization** and the total unseen **utilization** are accurately estimated according to the properties of one or more process service time distributions. The total unseen **utilization** is also estimated with an iterative method using gradations of the sample interval. The length distribution of the unseen processes is determined with the same iterative method.

Application Filing Date - AD (1):

19990407

TITLE - TI (1):

Enterprise management system and method which includes statistical recreation of system **resource** usage for more accurate **monitoring,** prediction, and performance workload characterization

Brief Summary Text - BSTX (3):

The present invention relates to the collection, analysis, and management of system **resource** data in distributed or enterprise computer systems, and particularly to the more accurate **monitoring** of the state of a computer system and more accurate prediction of system performance.

Brief Summary Text - BSTX (5):

The data processing resources of business organizations are increasingly taking the form of a distributed computing environment in which data and processing are dispersed over a network comprising many interconnected, heterogeneous, geographically remote computers. Such a computing environment is commonly referred to as an enterprise computing environment, or simply an enterprise. Managers of the enterprise often employ software packages known as enterprise management systems to **monitor,** analyze, and manage the **resources** of the enterprise. Enterprise management systems may provide for the collection of measurements, or metrics, concerning the resources of individual systems. For example, an enterprise management system might include a software agent on an individual computer system for the **monitoring** of particular **resources** such as CPU usage or disk access. U.S. Pat. No. 5,655,081 discloses one example of an enterprise management system.

Brief Summary Text - BSTX (6):

In a sophisticated enterprise management system, tools for the analysis, modeling, planning, and prediction of system resource **utilization** are useful for assuring the satisfactory performance of one or more computer systems in the enterprise. Examples of such analysis and modeling tools are the "ANALYZE" and "PREDICT" components of "BEST/1 FOR DISTRIBUTED SYSTEMS" available from BMC Software, Inc. Such tools usually require the input of periodic measurements of the usage of resources such as central processing units (CPUs), memory, hard disks, network bandwidth, and the like. To ensure accurate analysis and modeling, therefore, the collection of accurate performance data is critical.

Brief Summary Text - BSTX (13):

Sampled metric data can be used to build performance models for analysis and capacity planning. However, less frequent sampling can result in inaccurate models and data uncertainty, especially regarding the duration of events or processes and the number of events or processes. The present invention is directed to reducing said uncertainty. Uncertainty arises from two primary sources: the unsampled segment of a seen process or event, and the unseen process or event. A seen process is a process that is sampled at least once; therefore, its existence and starting time are known. However, the residual time or **utilization** between the last sampling of the process or event and the death of the process or the termination of the event is unsampled and unknown. An unseen process is shorter than the sample interval and is not sampled at all, and therefore its entire **utilization** is unknown. Nevertheless, the total unsampled (i.e., residual) **utilization** and the total unseen **utilization** can be estimated with the system and method of the present invention.

Brief Summary Text - BSTX (14):

In determining the total unsampled **utilization,** a quantity of process service time distributions are determined, and each of the seen processes are assigned respective process service time distributions. For each distribution, a mean residual time is calculated using equations provided by the system and method. The total unsampled **utilization** is the sum of the mean residual time multiplied by the number of seen processes for each distribution, all divided by the measurement interval.

Brief Summary Text - BSTX (15):
In determining the total unseen **utilization,** first the total captured **utilization** is determined to be the sum of the sampled **utilizations** of all seen processes over the measurement interval. Next the total measured **utilization,** or the "actual" **utilization** over the measurement interval, is obtained from the system software or monitoring software. The difference between the total measured **utilization** and the total captured **utilization** is the uncertainty. Because the uncertainty is due to either unsampled segments or unseen events, the total unseen **utilization** is calculated to be the uncertainty (the total measured **utilization** minus the total captured **utilization**) minus the total unsampled **utilization**.

Brief Summary Text - BSTX (16):
When the total measured **utilization** is not available, the total unseen **utilization** is estimated with an iterative bucket method. A matrix of buckets are created, wherein each row corresponds to the sample interval and each bucket to a gradation of the sample interval. Each process is placed into the appropriate bucket according to how many times it was sampled and when in the sample interval it began. Starting with the bucket with the longest process(es) and working iteratively back through the other buckets, the number of unseen processes are estimated for each length gradation of the sample interval. The iterative bucket method is also used to determine a length distribution of unseen processes.

Brief Summary Text - BSTX (17):
In response to the determination of **utilizations** described above, the system and method are able to use this information in modeling and/or analyzing the enterprise. In various embodiments, the modeling and/or analyzing may further comprise one of more of the following: displaying the determinations to a user, predicting future performance, graphing a performance prediction, generating reports, asking a user for further data, permitting a user to modify a model of the enterprise, and altering a configuration of the enterprise in response to the determinations.

Drawing Description Text - DRTX (14):
FIG. 12 is a flowchart illustrating the determination of the total uncaptured **utilization**;

Drawing Description Text - DRTX (15):
FIG. 13 is a flowchart further illustrating the determination of the total uncaptured **utilization**;

Drawing Description Text - DRTX (16):
FIG. 14 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for an exponential distribution;

Drawing Description Text - DRTX (17):
FIG. 15 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for a uniform distribution;

Drawing Description Text - DRTX (18):
FIG. 16 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for an unknown distribution;

Drawing Description Text - DRTX (19):
FIG. 17 is a flowchart illustrating an alternative method of the determination of the portion of the total uncaptured **utilization** for an unknown distribution;

Drawing Description Text - DRTX (20):
FIG. 18 is a flowchart illustrating the determination of the total unseen **utilization**;

Drawing Description Text - DRTX (21):
FIG. 19 illustrates a matrix of buckets used in the estimation of the total unseen **utilization**;

Drawing Description Text - DRTX (22):
FIG. 20 illustrates a specific example of the estimation of the total unseen **utilization** with buckets;

Drawing Description Text - DRTX (23):
FIG. 21 is a flowchart illustrating the iterative bucket method of estimating the total unseen **utilization**;

Detailed Description Text - DETX (2):
U.S. Pat. No. 5,655,081 titled "System for **Monitoring** and Managing Computer **Resources** and Applications Across a Distributed Environment Using an Intelligent. Autonomous Agent Architecture" is hereby incorporated by reference as though fully and completely set forth herein.

Detailed Description Text - DETX (10):
When the computer programs are executed on one or more computer systems 150, an enterprise management system 180 is operable to **monitor,** analyze, and manage the computer programs, processes, and **resources** of the enterprise 100. Each computer system 150 in the enterprise 100 executes or runs a plurality of software applications or processes. Each software application or process consumes a portion of the resources of a computer system and/or network: for example, CPU time, system memory such as RAM, nonvolatile memory such as a hard disk, network bandwidth, and input/output (I/O). The enterprise management

system 180 permits users to **monitor,** analyze, and manage **resource** usage on heterogeneous computer systems 150 across the enterprise 100.

Detailed Description Text - DETX (14):
FIG. 4 shows an overview of the Monitor component 402 of the console node 400 of the enterprise management system 180. The Monitor 402 comprises a Manager Daemon 430, one or more Monitor Consoles (as illustrated, 420a and 420b), and a **Policy** Registration Queue 440. Although two Monitor Consoles 420a and 420b are shown in FIG. 4, the present invention contemplates that one or more Monitor Consoles may be executing on any of one or more console nodes 400.

Detailed Description Text - DETX (15):
In the preferred embodiment, the Monitor Consoles 420a and 420b use a graphical user interface (GUI) for user input and information display. Preferably, the Monitor Consoles 420a and 420b are capable of sending several different types of requests to an Agent 302, including: alert requests, update requests, graph requests, and drilldown requests. An alert request specifies one or more **thresholds** to be checked on a routine basis by the Agent 302 to detect a problem on the agent node 300. For example, an alert request might ask the Agent 302 to report to the Monitor Console 420a whenever usage of a particular software process exceeds a particular **threshold** relative to overall CPU usage on the agent node 300. An update request is a request for the status of the Agent 302. For example, the requested status information might include the version number of the Agent 302 or the presence of any alarms in the Agent 302. A graph request is a request to receive graph data, i.e., data on a metric as routinely collected by the Agent 302, and to receive the data in real time, i.e., whenever it becomes available from the present time onward. By obtaining and displaying graph data, the Monitor Console 420a enables the rapid identification and communication of potential application and system performance problems. Preferably, the Monitor Console 420a displays graph data in a graphical format. A drilldown request is a request to receive drilldown data, i.e., data on an entire metric group (a set of metrics) as collected by the Agent 302. By obtaining and displaying drilldown data, the Monitor Console 420a provides the ability to focus, in real-time, on a specific set of processes, sessions, or users. Preferably, the Monitor Console 420a displays drilldown data in a tabular format.

Detailed Description Text - DETX (16):
Whenever the Agent 302 generates an alarm to indicate a troublesome status on the agent node 300, the Manager Daemon 430 intercepts the alarm and feeds the alarm to one or more Monitor Consoles, such as 420a and 420b. Typically, an alarm is a notification that a particular **threshold** has been exceeded on a monitored process or subsystem on an agent node 300. The Manager Daemon 430 is capable of receiving alarms from a plurality of Agents 302. A Manager Daemon 430 is preferably always running on each console node 400 so that alarms can be captured even when the Monitor Consoles 420a and 420b are offline.

Detailed Description Text - DETX (17):
Each of the Monitor Consoles 420a and 420b is able to issue one or more

policies. A **policy** defines a disparate set of metrics to be collected on one or more agent nodes 300. In other words, a **policy** allows a Monitor Console 420a or 420b to monitor one or more metrics on one or more agent nodes 300 simultaneously. For example, a user could build and deploy a **policy** that restricts web browser access on a plurality of agent nodes 300 with the following set of interrelated conditions: "IF more than 80% of server CPU is required by critical production applications, AND the run queue length is greater than six, AND active time on production disks exceeds 40%." **Policies** are registered with the **Policy** Registration Queue 440, from which they are disseminated to the appropriate Agents 302. An Agent 302 can execute a plurality of **policies** simultaneously.

Detailed Description Text - DETX (24):

FIG. 6 illustrates an overview of the Analyze component 406 of the console node 400 of the enterprise management system 180. In the preferred embodiment, Analyze 406 comprises the "ANALYZE" portion of the "BEST/1 FOR DISTRIBUTED SYSTEMS" software package available from BMC Software, Inc. Essentially, Analyze 406 takes the data collected by one or more Agents. 302 and creates a model of one or more computer systems and the processes that run on those computer systems. In the preferred embodiment, Analyze 106 can model multi-vendor environments, system memory, multiple processors, disk drives, logical volumes, RAID devices, load **balancing**, ASCII and X terminals, local and remote file servers, independent and dependent transactions, client/server workloads, private and shared memory/transaction, CPU priority scheduling, networks of different types, and "ORACLE", "SYBASE", and "INFORMIX" database environments. In the preferred embodiment, Analyze 406 takes as input a domain file 466 which identifies the agent nodes 300 on the network and the relationship between them. As shown in FIG. 6, Analyze 406 also takes as input a data repository in either UDF 212c or UDR 210c format, wherein the data repository 212c or 210c is a set of metric groups collected from one or more agent nodes 300.

Detailed Description Text - DETX (27):

FIG. 7 shows an overview of the Predict component 408 of the console node 400 of the enterprise management system 18.0. In the preferred embodiment, Predict 408 comprises the "BEST/1-PREDICT" component of the "BEST/1 FOR DISTRIBUTED SYSTEMS" software package available from BMC Software, Inc. Predict 408 is a planning tool which forecasts the impact of hypothetical changes on elements of the enterprise 100 such as disparate hardware, software, applications, and databases. Predict 408 takes the workload data from a Model File 468c, such as the Model File 468a generated by Analyze 406, and computes performance statistics such as workload response times, **utilization**, and throughputs at CPUs, disks, networks, and other elements of the enterprise computing environment 100. Thus, Predict 408 constructs a baseline model from collected data that represents the essence of the system under management. The user can also operate Predict 408 to construct the baseline model from pre-built model components, or from a combination of collected data and pre-built components. Preferably, Predict 408 uses a graphical user interface (GUI) for user input and information display.

Detailed Description Text - DETX (37):

The events being sampled may include, for example, process lifetimes, process types, or disk access times, or any other performance metrics that can be monitored. Although this description addresses in detail examples such as CPU **utilization**, process lifetime, and process type, the system and method can be applied to any metric. As used herein, "process" refers to an executing program, a task, a thread, or any other unit of execution.

Detailed Description Text - DETX (40):

Uncertainty arises from two primary sources: the unsampled segment of a seen event or process, and the unseen, short-lived event or process. FIG. 9 is a diagram illustrating an unsampled segment of a seen event. The horizontal line designated "Time" indicates increasing time from left to right. The timeline encompasses all or part of the measurement interval L. The vertical lines labeled s.sub.(il)-2 through s.sub.(im,)+3 indicate samples taken at a constant sample interval .DELTA.. The event or process 610 begins at the point in time b.sub.i and ends at the point in time d.sub.i. The process 610 begins after sample s.sub.(il)-1 but before sample s.sub.il, so the process 610 is not detected at the point in time b.sub.i when it begins. However, the process 610 is still executing when sample s.sub.il is taken, so the existence of this process 610 is known at that point. In other words, the process 610 is a seen process or a known process as soon as the first sample s.sub.il is taken. Furthermore, in a preferred embodiment, the starting time b.sub.i of the process 610 is also determined when the process 610 is detected at sample s.sub.il. After it has first been sampled, the process 610 continues executing for an indefinite period of time, as indicated in FIG. 9 by broken lines, wherein the process lifetime may or may not encompass additional samplings of the process 610 at regular sample intervals. The last sampling of the process 610, and therefore the last time the process 610 is seen, is the m.sub.i th sample at the point in time s.sub.im.sub..sub.i . The present invention contemplates that a seen process may be sampled only once, and thus that s.sub.il =s.sub.im.sub..sub.i in some cases. The process 610 stops executing at the point in time d.sub.i, after s.sub.im.sub..sub.i but prior to s.sub.(im.sub..sub.i .sub.)+1. In the preferred embodiment, however, no record is kept of the termination of the process 610, and so the length of the process 610 after s.sub.im.sub..sub.i is unknown. Therefore, the known, captured, or sampled length 612 of the seen process 610 is represented by the difference between s.sub.im.sub..sub.i and b.sub.i. The unsampled or unknown length 614 of the seen process 610 is represented by the difference between d.sub.i and s.sub.im.sub..sub.i . The unsampled segment 614 is also known as the residual process time. The captured **utilization** is the sampled length 612 divided by the measurement interval L.

Detailed Description Text - DETX (42):

For a computer system or plurality of systems with hundreds or thousands of processes starting and ending within a measurement interval, the uncertainty adds up rapidly and can distort a performance model. However, statistical methods according to the present invention can provide estimations of the uncertain data, thus recreating the lost data and reducing uncertainty. FIG. 11 is a flowchart illustrating an overview of the statistical estimation of metric data. The difference between the "actual" total **utilization** and the

"sampled" total **utilization**--in other words, the uncertainty--can be distributed both to the unsampled segments of the seen events or processes and to the unseen events or processes. Accordingly, in step 720 of FIG. 11 the total uncaptured **utilization** U.sub.uc is estimated. U.sub.uc represents an estimate of the total unsampled **utilization** of all seen processes over the measurement interval L. In step 722 the total unseen **utilization** U.sub.us is estimated. U.sub.us represents an estimate of the total **utilization** of all unseen processes for the measurement interval L.

Detailed Description Text - DETX (43):
FIG. 12 is a flowchart illustrating the determination of the total uncaptured **utilization** U.sub.uc. In step 738 the measurement interval L is determined. The steps thereafter are performed for measurements within the interval L. In step 740 one or more process service time distributions are determined, wherein the quantity of distributions is labeled d. A process service time distribution is a statistical distribution which determines the duration of one or more processes. In step 742 the quantity n.sub.cp of seen processes which follow each distribution is determined. In other words, in steps 740 and 742 the seen processes are divided into d groups, wherein each group represents processes that are characterized by the same process service time distribution.

Detailed Description Text - DETX (45):
In step 746 the total uncaptured **utilization** U.sub.uc is determined according to the following equation: ##EQU1##

Detailed Description Text - DETX (46):
wherein d is the number of process service time distributions, r.sub.j is the mean residual time for each distribution j, n.sub.cpj is the number of seen processes for each distribution j, and L is the measurement interval. In other words, the total uncaptured **utilization** U.sub.uc is the sum of the products of the mean residual time and the number of seen processes for each distribution, all divided by the measurement interval. If there is only one process service time distribution, however, then the total uncaptured **utilization** U.sub.uc can be determined according to a simplified equation: ##EQU2##

Detailed Description Text - DETX (48):
FIG. 13 illustrates the general determination of the total uncaptured **utilization** U.sub.uc for any process service time distribution. In step 750 the process service time distribution is determined. In step 752 the quantity n.sub.cp of seen processes which follow this distribution is determined.

Detailed Description Text - DETX (53):
FIG. 14 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for an exponential distribution. In step 760 the process service time distribution is determined to be an exponential distribution with service rate .lambda.. In step 762 the quantity n.sub.cp of seen processes which follow the exponential distribution with service rate

.lambda. is determined. In step 764 the mean residual time r for the exponential distribution is determined according to the following equation: ##EQU5##

Detailed Description Text - DETX (55):
FIG. 15 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for a uniform distribution. In step 780 the process service time distribution is determined to be a uniform distribution between zero and a constant C. In step 782 the quantity n.sub.cp of seen processes which follow the uniform distribution between zero and C is determined. In step 784 the mean residual time r for the uniform distribution is determined according to the following equation: ##EQU6##

Detailed Description Text - DETX (57):
FIG. 16 is a flowchart illustrating the determination of the portion of the total uncaptured **utilization** for an unknown distribution. In step 800 the process service time distribution is determined to be an unknown distribution. In step 802 the quantity n.sub.cp of seen processes which follow the unknown distribution is determined. In step 804 the mean residual time r for the unknown distribution is determined according to the following equation: ##EQU7##

Detailed Description Text - DETX (59):
FIG. 17 is a flowchart illustrating an alternative method of the determination of the portion of the total uncaptured **utilization** for an unknown distribution. In step 820 the process service time distribution is determined to be an unknown distribution. In step 822 the quantity n.sub.cp of seen processes which follow the unknown distribution is determined. In step 824 the mean residual time r for the unknown distribution is determined according to the following equation: ##EQU8##

Detailed Description Text - DETX (61):
FIG. 18 is a flowchart illustrating the determination of the total unseen **utilization**. In step 840 a total captured **utilization** U.sub.c is determined. The total captured **utilization** U.sub.c is the sum of the sampled lengths of all seen processes over the measurement interval L and can be computed as follows: ##EQU9##

Detailed Description Text - DETX (63):
In step 842 a total measured **utilization** U.sub.m is determined. The total measured **utilization** represents the total **utilization** of all processes of interest, seen and unseen, over the measurement interval L. Step 842 assumes that universal **utilization** statistics are available from the registry of metrics, system software, or other monitoring software. In step 844 the total unseen **utilization** U.sub.us is determined according to the following equation:

Detailed Description Text - DETX (64):

In other words, the uncertainty is the difference between the "actual" **utilization** U.sub.m and the "sampled" **utilization** U.sub.c. The uncertainty (i.e., U.sub.m -U.sub.c) is the sum of the uncaptured **utilization** U.sub.uc and the unseen **utilization** U.sub.us, so the unseen **utilization** U.sub.us can be determined once the uncaptured **utilization** U.sub.uc and the uncertainty are known. As discussed above, FIG. 12 illustrates how the uncaptured **utilization** U.sub.uc can be computed in one embodiment.

Detailed Description Text - DETX (65):
If, however, universal **utilization** statistics are not available, then the total measured **utilization** U.sub.m cannot easily be determined. Nevertheless, the total unseen **utilization** U.sub.us may still be determined according to an iterative method which is described as follows. The following method is also useful when the processes of interest do not represent all the activity on the computer system. If so, it is assumed that the processes are marked in such a way that it can be determined which ones are of interest. Furthermore, it is assumed in all cases that the start time of processes is independent of the sample time. Therefore, if there are very many short-lived processes, then a certain percentage of them will be seen. Because the percentage of short-lived, seen processes can be estimated as described below, the total number of unseen processes can be estimated as well.

Detailed Description Text - DETX (92):
The iterative technique described above is useful even when the total measured **utilization** U.sub.m is known. With this technique, the length distribution of unseen processes can be determined. The distribution should be proportional to the number of unseen processes in each bucket. Let U.sub.us(i) be the **utilization** of unseen processes of length between ##EQU32##

Detailed Description Text - DETX (93):
and e.sub.max, and let U.sub.us(j) be the **utilization** of unseen processes of length between ##EQU33##

Detailed Description Text - DETX (95):
wherein j=0,1,2,. . . , i-1. The **utilization** for the unseen processes U.sub.us can be distributed as: ##EQU35##

Detailed Description Text - DETX (96):
wherein j=0,1,2,. . . , i-1. The two equations are illustrated in FIGS. 22 and 23, respectively. In other words, the length distribution of unseen processes is determined by multiplying the total unseen **utilization** U.sub.us by a coefficient, wherein the coefficient and is derived from the iterative method.

Detailed Description Text - DETX (97):
In one embodiment, the enterprise is modeled and/or its configuration is altered in response to the determination(s) of **utilization** described herein.

Modeling according to one embodiment is discussed in detail with reference to FIGS. 6 and 7. In various embodiments, this modeling may further comprise one of more of the following: displaying the determination(s) to a user, predicting future performance, graphing a performance prediction, generating reports, asking a user for further data, and permitting a user to modify a model of the enterprise. In one embodiment, Analyze 406 and/or Predict 408, as discussed in detail with reference to FIGS. 6 and 7, implement the modeling, analysis, and/or prediction in response to the determination(s) of **utilization**. In one embodiment, a configuration of the enterprise is altered in response to the determination(s) of **utilization**. Altering a configuration of the enterprise may comprise, for example, reconfiguring a network topology or installing additional resources, such as CPUs, software, memory resources, or network routers or hubs.

Claims Text - CLTX (1):
1. A method for monitoring the state of a computer system, the method comprising: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to one or more processes on the computer system; storing the set of raw data points in a memory; sampling the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes of the one or more processes over the measurement interval; statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes of the one or more processes over the measurement interval.

Claims Text - CLTX (2):
2. The method of claim 1, wherein the statistically estimating a total uncaptured **utilization** U.sub.uc further comprises: determining a process service time distribution, wherein the process service time distribution estimates a duration of one or more processes; determining a quantity n.sub.cp of seen processes which follow the process service time distribution; determining a mean residual time r for the process service time distribution, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process; determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU36## wherein L is the measurement interval.

Claims Text - CLTX (8):
8. The method of claim 1, wherein the statistically estimating a total uncaptured **utilization** U.sub.uc further comprises: determining a plurality d of process service time distributions, wherein each process service time distribution j estimates a duration of one or more processes, wherein 1.ltoreq.j.ltoreq.d; for each process service time distribution j, determining

a quantity n.sub.cpj of seen processes which follow that process service time distribution j; for each process service time distribution j, determining a mean residual time r.sub.j for that process service time distribution j, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process which follows that process service time distribution j; determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU43## wherein L is the measurement interval.

Claims Text - CLTX (14):

14. The method of claim 1, further comprising: determining a total captured **utilization** U.sub.c, wherein the total captured **utilization** measures a total length of sampled segments for the one or more seen processes over the measurement interval; determining a total measured **utilization** U.sub.m, wherein the total measured **utilization** U.sub.m measures a total length of all of the one or more processes over the measurement interval; wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises determining the total unseen **utilization** U.sub.us according to the following equation:

Claims Text - CLTX (15):

wherein U.sub.uc is the total uncaptured **utilization**.

Claims Text - CLTX (16):

15. The method of claim 14, wherein each process has a beginning time, and wherein the set of sampled data points includes the beginning time b.sub.i for each seen process; wherein the determining a total captured **utilization** U.sub.c further comprises determining U.sub.c according to the following equation: ##EQU50## wherein CP is a set of all seen processes, s.sub.im.sub..sub.i is the last sampling time for each seen process i.epsilon.CP, b.sub.i is the beginning time for each seen process i.epsilon.CP, and L is the measurement interval.

Claims Text - CLTX (17):

16. The method of claim 1, wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each segment corresponds to a bucket.

Claims Text - CLTX (18):

17. The method of claim 1, wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises: creating a plurality of buckets with m rows and n columns, wherein n is a maximum number of samples in the set of sampled data points for any particular process, wherein m is a multiple of n, and wherein the buckets are ordered from zero to m-1; placing each seen process into one of the plurality of buckets, wherein the bucket is labeled

according to the following equation: ##EQU51## wherein t is a total quantity of samples in the set of sampled data points for this process, wherein i indicates one of ##EQU52##

Claims Text - CLTX (21):

18. The method of claim 1, further comprising: determining a length distribution of the unseen processes of a greatest length, comprising multiplying the total unseen **utilization** U.sub.us by a first coefficient; determining a length distribution of the unseen processes of a lesser length, comprising multiplying the total unseen **utilization** U.sub.us by a second coefficient; wherein the first coefficient and second coefficient are derived from an iterative method, wherein the iterative method comprises: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each length segment corresponds to a bucket.

Claims Text - CLTX (25):

22. The method of claim 1, wherein the collecting a set of raw data points, the storing the set of raw data points in a memory, the sampling the memory, the statistically estimating a total uncaptured **utilization** U.sub.uc, and the statistically estimating a total unseen **utilization** U.sub.us are performed on a single computer system.

Claims Text - CLTX (26):

23. The method of claim 1, wherein the collecting a set of raw data points is performed on a different computer system than the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (27):

24. The method of claim 1, further comprising: modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (28):

25. The method of claim 1, further comprising: altering a configuration of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (29):

26. A method for monitoring the state of a computer system, the method comprising: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to one or more processes on the computer system; storing the set of raw data points in a memory; sampling the

memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes of the one or more processes over the measurement interval, comprising: determining a plurality d of process service time distributions, wherein each process service time distribution j estimates a duration of one or more processes, wherein 1.ltoreq.j.ltoreq.d; for each process service time distribution j, determining a quantity n.sub.cpj of seen processes which follow that process service time distribution j; for each process service time distribution j, determining a mean residual time r.sub.j for that process service time distribution j, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process which follows that process service time distribution j; determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU55## statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes of the one or more processes over the measurement interval; modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (30):
27. A method for monitoring the state of a computer system, the method comprising: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to one or more processes on the computer system; storing the set of raw data points in a memory; sampling the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes over the measurement interval; statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes over the measurement interval, comprising: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each segment corresponds to a bucket; modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (31):
28. A system for monitoring the state of a computer system, the system comprising: a CPU; a system memory coupled to the CPU, wherein the system

memory stores one or more computer programs executable by the CPU; wherein the computer programs are executable to: collect a set of raw data points over a measurement interval L, wherein the set of raw data points relates to a set of processes on the computer system; store the set of raw data points in a memory; sample the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimate a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes over the measurement interval; statistically estimate a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes over the measurement interval.

Claims Text - CLTX (32):

29. The system of claim 28, wherein in statistically estimating the total uncaptured **utilization** U.sub.uc, the computer programs are executable to: determine a process service time distribution, wherein the process service time distribution estimates a duration of one or more processes; determine a quantity n.sub.cp of seen processes which follow the process service time distribution; determine a mean residual time r for the process service time distribution, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process; determine the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU56## wherein L is the measurement interval.

Claims Text - CLTX (38):

35. The system of claim 28, wherein in statistically estimating a total uncaptured **utilization** U.sub.uc, the computer programs are executable to: determine a set d of process service time distributions, wherein each process service time distribution j estimates a duration of one or more processes, wherein 1.ltoreq.j.ltoreq.d; for each process service time distribution j, determine a quantity n.sub.cpj of seen processes which follow that process service time distribution j; for each process service time distribution j, determine a mean residual time r.sub.j for that process service time distribution j, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process which follows that process service time distribution j; determine the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU63## wherein L is the measurement interval.

Claims Text - CLTX (45):

41. The system of claim 28, wherein the computer programs are further executable to: determine a total captured **utilization** U.sub.c, wherein the total captured **utilization** measures a total length of sampled segments for one or more seen processes of over the measurement interval; determine a total measured **utilization** U.sub.m, wherein the total measured **utilization** U.sub.m measures a total length of all of the one or more processes over the

measurement interval; wherein in statistically estimating a total unseen **utilization** U.sub.us, the computer programs are executable to determine the total unseen **utilization** U.sub.us according to the following equation:

Claims Text - CLTX (46):

42. The system of claim 41, wherein each process has a beginning time, and wherein the set of sampled data points includes the beginning time b.sub.i for each seen process; wherein in determining a total captured **utilization** U.sub.c, the computer programs are executable to determine U.sub.c according to the following equation: ##EQU70## wherein CP is a set of all seen processes, S.sub.im.sub..sub.i is the last sampling time for each seen process i.epsilon.CP, b.sub.i is the beginning time for each seen process i.epsilon.CP, and L is the measurement interval.

Claims Text - CLTX (47):

43. The system of claim 28, wherein in statistically estimating a total unseen **utilization** U.sub.us, the computer programs are executable to: create a plurality of buckets; place each seen process into one of the plurality of buckets; estimate a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each segment corresponds to a bucket.

Claims Text - CLTX (48):

44. The system of claim 28, wherein in statistically estimating a total unseen **utilization** U.sub.us, the computer programs are executable to: create a plurality of buckets with m rows and n columns, wherein n is a maximum number of samples in the set of sampled data points for any particular process, wherein m is a multiple of n, and wherein the buckets are ordered from zero to m-1; place each seen process into one of the plurality of buckets, wherein the bucket is labeled according to the following equation: ##EQU71## wherein t is a total quantity of samples in the set of sampled data points for this process, wherein i indicates one of ##EQU72##

Claims Text - CLTX (50):

45. The system of claim 28, wherein the computer programs are further executable to: determine a length distribution of the unseen processes of a greatest length, comprising multiplying the total unseen **utilization** U.sub.us by a first coefficient; determine a length distribution of the unseen processes of a lesser length, comprising multiplying the total unseen **utilization** U.sub.us by a second coefficient; wherein the first coefficient and second coefficient are derived from an iterative method, wherein in performing the iterative method to determine the first coefficient and second coefficient, the computer programs are executable to: create a plurality of buckets; place each seen process into one of the plurality of buckets; estimate a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each length segment corresponds to a bucket.

Claims Text - CLTX (52):

47. The system of claim 28, wherein the computer programs are further executable to modify a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (53):

48. The system of claim 28, wherein the computer programs are further executable to alter a configuration of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (54):

49. A memory medium which stores program instructions for monitoring the state of a computer system, wherein the program instructions are executable to implement: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to a one or more processes on the computer system; storing the set of raw data points in a memory; sampling the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes over the measurement interval; statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes over the measurement interval.

Claims Text - CLTX (55):

50. The memory medium of claim 49, wherein the statistically estimating a total uncaptured **utilization** U.sub.uc further comprises: determining a process service time distribution, wherein the process service time distribution estimates a duration of one or more processes; determining a quantity n.sub.cp of seen processes which follow the process service time distribution; determining a mean residual time r for the process service time distribution, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process; and determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU74## wherein L is the measurement interval.

Claims Text - CLTX (61):

56. The memory medium of claim 49, wherein the statistically estimating a total uncaptured **utilization** U.sub.uc further comprises: determining a set d of process service time distributions, wherein each process service time distribution j estimates a duration of one or more processes, wherein 1.ltoreq.j.ltoreq.d; for each process service time distribution j, determining a quantity n.sub.cpj of seen processes which follow that process service time

distribution j; for each process service time distribution j, determining a mean residual time r.sub.j for that process service time distribution j, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process which follows that process service time distribution j; and determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU81## wherein L is the measurement interval.

Claims Text - CLTX (67):

62. The memory medium of claim 49, wherein the program instructions further implement: determining a total captured **utilization** U.sub.c, wherein the total captured **utilization** measures a total length of sampled segments for the one or more seen processes over the measurement interval; and determining a total measured **utilization** U.sub.m, wherein the total measured **utilization** measures a total length of all of the one or more processes over the measurement interval; wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises determining the total unseen **utilization** U.sub.us according to the following equation:

Claims Text - CLTX (68):

63. The memory medium of claim 62, wherein each process has a beginning time, and wherein the set of sampled data points includes the beginning time b.sub.i for each seen process; wherein the determining a total captured **utilization** U.sub.c further comprises determining U.sub.c according to the following equation: ##EQU88## wherein CP is a set of all seen processes, s.sub.im.sub..sub.i is the last sampling time for each seen process i.epsilon.CP, b.sub.i is the beginning time for each seen process i.epsilon.CP, and L is the measurement interval.

Claims Text - CLTX (69):

64. The memory medium of claim 49, wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; and estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each segment corresponds to a bucket.

Claims Text - CLTX (70):

65. The memory medium of claim 49, wherein the statistically estimating a total unseen **utilization** U.sub.us further comprises: creating a plurality of buckets with m rows and n columns, wherein n is a maximum number of samples in the set of sampled data points for any particular process, wherein m is a multiple of n, and wherein the buckets are ordered from zero to m-1; placing each seen process into one of the plurality of buckets, wherein the bucket is labeled according to the following equation: ##EQU89## wherein t is a total quantity of samples in the set of sampled data points for this process, wherein i indicates one of ##EQU90##

Claims Text - CLTX (74):

66. The memory medium of claim 49, wherein the program instructions further implement: determining a length distribution of the unseen processes of a greatest length, comprising multiplying the total unseen **utilization** U.sub.us by a first coefficient; and determining a length distribution of the unseen processes of a lesser length, comprising multiplying the total unseen **utilization** U.sub.us by a second coefficient; wherein the first coefficient and second coefficient are derived from an iterative method, wherein the iterative method comprises: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; and estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each length segment corresponds to a bucket.

Claims Text - CLTX (78):

70. The memory medium of claim 49, wherein the collecting a set of raw data points, the storing the set of raw data points in a memory, the sampling the memory, the statistically estimating a total uncaptured **utilization** U.sub.uc, and the statistically estimating a total unseen **utilization** U.sub.us are performed on a single computer system.

Claims Text - CLTX (79):

71. The memory medium of claim 49, wherein the collecting a set of raw data points is performed on a different computer system than the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (80):

72. The memory medium of claim 49, wherein the programs instructions further implement modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (81):

73. The memory medium of claim 49, wherein the programs instructions further implement altering a configuration of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (82):

74. A memory medium which stores program instructions for monitoring the state of a computer system, wherein the program instructions are executable to implement: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to one or more processes on the computer system; storing the set of raw data points in a memory; sampling the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data

points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes of the one or more processes over the measurement interval, comprising: determining a set d of process service time distributions, wherein each process service time distribution j estimates a duration of one or more processes, wherein 1.ltoreq.j.ltoreq.d; for each process service time distribution j, determining a quantity n.sub.cpj of seen processes which follow that process service time distribution j; for each process service time distribution j, determining a mean residual time r.sub.j for that process service time distribution j, wherein the mean residual time estimates a length of an uncaptured residual segment for each seen process which follows that process service time distribution j; and determining the total uncaptured **utilization** U.sub.uc according to the following equation: ##EQU93## statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes of the one or more processes over the measurement interval; and modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Text - CLTX (83):

75. A memory medium which stores program instructions for monitoring the state of a computer system, wherein the program instructions are executable to implement: collecting a set of raw data points over a measurement interval L, wherein the set of raw data points relates to one or more processes on the computer system; storing the set of raw data points in a memory; sampling the memory repetitively at a sample interval .DELTA. to create a set of sampled data points, wherein processes which are included in the set of sampled data points are seen processes and processes which are not included in the set of sampled data points are unseen processes, and wherein the set of sampled data points includes a first sampling time and a last sampling time for each seen process; statistically estimating a total uncaptured **utilization** U.sub.uc, wherein the total uncaptured **utilization** is an estimation of a total length of unsampled segments for the seen processes over the measurement interval; and statistically estimating a total unseen **utilization** U.sub.us, wherein the total unseen **utilization** is an estimation of a total length of the unseen processes over the measurement interval, comprising: creating a plurality of buckets; placing each seen process into one of the plurality of buckets; estimating a total quantity of unseen processes for each of a plurality of equal length segments of the sample interval .DELTA., wherein each segment corresponds to a bucket; and modifying a model of the computer system based on the statistically estimating a total uncaptured **utilization** U.sub.uc and the statistically estimating a total unseen **utilization** U.sub.us.

Claims Paragraph Equation - CLEQ (2):

U.sub.us =U.sub.m -U.sub.c -U.sub.uc, wherein U.sub.uc is the total uncaptured **utilization.**

Claims Paragraph Equation - CLEQ (3):

U.sub.us =U.sub.m -U.sub.c U.sub.uc, wherein U.sub.uc is the total uncaptured **utilization.**